

Transform Domain Analysis of Sequences

Muhammad Asad Khan, Fauzan Mirza, and Amir Ali Khan

National University of Sciences and Technology,
Islamabad, Pakistan
asad.khan@seecs.edu.pk

Abstract. In cryptanalysis, security of ciphers vis-a-vis attacks is gauged against three criteria of complexities, i.e., computations, memory and time. Some features may not be so apparent in a particular domain, and their analysis in a transformed domain often reveals interesting patterns. Moreover, the complexity criteria in different domains are different and performance improvements are often achieved by transforming the problem in an alternate domain. Owing to the results of coding theory and signal processing, Discrete Fourier Transform (DFT) based attacks have proven to be efficient than algebraic attacks in terms of their computational complexity. Motivated by DFT based attacks, we present a transformed domain analysis of Linear Feedback Shift Register (LFSR) based sequence generators. The time and frequency domain behavior of nonlinear filter and combiner generators is discussed along with some novel observations based on the Chinese Remainder Theorem (CRT). CRT is exploited to establish patterns in LFSR sequences and underlying cyclic structures of finite fields. Application of DFT spectra attacks on combiner generators is also demonstrated. Our proposed method saves on the last stage computations of selective DFT attacks for combiner generators. The proposed approach is demonstrated on some examples of combiner generators and is scalable to general configuration of combiner generators.

Keywords: LFSR, DFT, filter generator, combiner generator, CRT.

1 Introduction

LFSRs have been widely used for sequence generation due to their inbuilt recursive structure, faster implementations and well studied behaviour in diverse applications of communications, coding theory and cryptology. In cryptographic algorithms, the linear recurrence of LFSRs is modified by nonlinear filtering to achieve higher linear complexities and good statistical properties. Some of the classical schemes include filter generators, combiner generators, clock controlled generators and shrinking generators. The nonlinear function used in these generators is a boolean function $f : GF(2^n) \rightarrow GF(2)$ which takes l inputs from either an LFSR or from outputs of number of LFSRs and produces $GF(2)$ keystream sequence. LFSR based sequence generators serve as the basic building blocks for a number of e-stream submissions [1] and cellular algorithms i.e. A5 series [2] and E0 algorithm [6].

In many scientific applications, it is common to analyze a problem in different domains because some characteristics of the involved parameters can only be better revealed in a particular domain. Moreover, the computational requirements can be reduced as result of the analysis in some transformed domains. For instance, the propagation characteristics of light or magnetic waves can be conveniently observed in frequency domain as compared to its equivalent representation in time domain. A continuous signal in time domain $s(t) = \cos(2\pi f_c t)$ having infinite non-zero points once converted to frequency domain $S(f)$ has only two non-zero frequency components i.e. f_c and $-f_c$. Similarly decoding of Block codes like Bose, Chaudhuri, and Hocquenghem (BCH) codes and Reed Solomon codes in frequency domain have proven to be more efficient in terms of computational complexity and error free recovery as compared to its analogous time domain decoding techniques [28]. Contemplating the same, analysis of cryptographic primitives in transform domains has also produced promising results. Though very less known in public literature, DFT based spectral attacks [19] and transform domain analysis of DES cipher components [18] are interesting examples in this regard.

In this report, analysis of LFSR based sequence generators in time and frequency domains has been presented. Starting from time and frequency domain analysis of basic LFSR sequences, we build our analysis onto filter and combiner generators. In Section-2, basics of Fourier transform over finite fields is recalled. Section-3 delineates the time and frequency domain analysis of LFSRs. Section-4 describes transform domain analysis of a simple product sequence which is fundamental component of non-linear boolean functions. A novel account of Chinese Remainder Theorem (CRT) based interpretation of fixed patterns in cyclic structures of underlying finite fields is discussed in this section. In Section-5, time and frequency domain analysis of non-linear filter generators is given with a perspective of their application in cryptographic algorithms. Specific comments about selective DFT attacks on filter generators [19] are specifically made in this section. Section-6 discusses transform domain analysis of combiner generators and application of selective DFT attacks on combiner generators by exploiting modular computations of CRT alongwith detailed complexity comparison in relation to classical divide and conquer attacks. In Section-7 discussion on applicability of discrete Fourier spectra attacks on A5/1 algorithm is made. The report is finally concluded in section-8.

2 Frequency Domain Representation over Finite Fields

Discrete Fourier Transform (DFT) is considered one of the most important discovery in the area of signal processing. DFT presents us with an alternate mathematical tool that allows us to examine the frequency domain behaviour of signals, often revealing important information not apparent in time domain. DFT S_k of an n -point sequence s_i is expressed in terms of inner product between the sequence and set of complex discrete frequency exponentials:

$$S_k = \sum_{i=0}^{n-1} s_i e^{-j2\pi ik/n}, \quad k = 0, 1, 2, \dots, n-1 \quad (1)$$

The term $e^{-j2\pi ik/n}$ represents discrete set of exponentials. Alternatively, $e^{-j2\pi/n}$ can be viewed as n^{th} root of unity.

Analogous to the classical DFT, a DFT for a periodic signal s_t with period n defined over a finite field $GF(2^m)$ is represented as

$$S_k = \sum_{t=0}^{n-1} s_t \alpha^{-tk}, \quad k = 0, 1, 2, \dots, n-1 \quad (2)$$

where S_k is k -th frequency component of DFT and α is the primitive element; generator of $GF(2^m)$ with period n [23]. For Inverse DFT, we will have a relation

$$s_t = \sum_{k=0}^{n-1} S_k \alpha^{tk}, \quad k = 0, 1, 2, \dots, n-1 \quad (3)$$

Similarly for polynomials, we have a relation for DFT and IDFT. Having a correspondence between a minimum polynomial and its associated sequence s_t with $s(x) = \sum_{t=0}^{n-1} s_t x^t$ and $S(x) = \sum_{k=0}^{n-1} S_k x^k$, following relation holds for DFT [15]:

$$S_k = s(\alpha^{-k}), \quad k = 0, 1, 2, \dots, n-1 \quad (4)$$

and similarly for IDFT:

$$s_t = S(\alpha^t), \quad t = 0, 1, 2, \dots, n-1 \quad (5)$$

3 Transformed Domain Analysis of LFSR Sequences

Classical theory on LFSR sequences and their applications in cryptology can be found in [16], [15] and [27]. In this section, transformed domain analysis of LFSRs, their sequences and underlying algebraic structures are recalled as they are fundamental to our proposed approach on filter and combiner generators.

3.1 Time Domain Representation of an LFSR Sequence

A binary sequence s_t can be treated as an LFSR sequence of degree m if it follows a linear recursion with coefficients from $GF(2)$ as:

$$s_{i+m} = \sum_{k=0}^{m-1} c_k s_{i+k} \quad \text{for } i \geq 0 \quad (6)$$

The value m is called the order of recurrence and associated characteristic polynomial in $GF(2)$ is defined by

$$f(x) = x^m + \sum_{k=0}^{m-1} c_k s_{i+k} \quad \text{for } i \geq 0 \quad (7)$$

The initial state $(s_0, s_1, \dots, s_{m-1})$ of an LFSR serves as a key to generate the complete sequence s_t . The period of any non-zero sequence can be at most $2^m - 1$ which is in relation to the characteristic polynomial $f(x)$ of the LFSR. If $f(x)$ is irreducible, it has m distinct roots i.e. α and its conjugate set $\{\alpha, \alpha^2, \alpha^4, \dots, \alpha^{2^{m-1}}\}$. Consequently, if $f(x)$ is a primitive polynomial, then order of its root α must be $2^m - 1$ which in other words is a period of associated sequence s_t . Thus a sequence s_t of an LFSR given by a primitive polynomial has maximum possible period $2^m - 1$ and is called m -sequence.

Trace Representation of an LFSR Sequence. The same sequence s_t can also be expressed in terms of its trace representation [24]; a linear operator from $GF(2^m)$ to its subfield $GF(2)$. Let $Tr_1^m(x) = \sum_{k=0}^{m-1} x^{2^k}$ be the trace mapping from $GF(2^m)$ to $GF(2)$, then m sequence s_t can be represented as:

$$s_t = Tr_1^m(\beta\alpha^t) \quad (8)$$

where α is a generator of a cyclic group $GF(2^m)^*$ and is called as primitive element of $GF(2^m)$. Note that $\beta \in GF(2^m)$ and each of its nonzero value corresponds to cyclic shift of the m -sequence generated by an LFSR with primitive polynomial $f(x)$. Importance of this interpretation of m -sequence is that different sequences constructed from root α of primitive polynomial $f(x)$ are cyclic shifts of the same m -sequence. The associated linear space $G(f)$ of dimension m contains 2^m different binary sequences including all 0s sequence as:

$$G(f) = \{ \tau^i s \mid 0 \leq i \leq 2^m - 2 \} \cup \{0\} \quad (9)$$

where τ is a left shift operator and represents a linear transformation of sequence s_t . It is important to mention here that all sequences in $G(f)$, defined over a primitive polynomial $f(x)$, have maximum period r i.e. $2^m - 1$ with an obvious exception of all 0s sequence. Moreover, any two sequences s and u within $G(f)$ are cyclic shift equivalent, if there exists an integer k such that

$$u_i = s_{i+k}, \quad \forall i \geq 0. \quad (10)$$

LFSR Sequence in Matrix Form. Each state of an m stage LFSR is a vector in the m -dimensional space $GF(2^m)$. The shift register is then a linear operator that changes the current state to its successor vector according to the feedback. In simple terms, the transformation of each non-zero sequence in a field, from state $(s_k, s_{k+1}, \dots, s_{k+m-1})$ to its successor state $(s_{k+1}, s_{k+2}, \dots, s_{k+m})$ can be regarded as a linear operation on $GF(2^m)$. The advantage of working with operator operating on m -dimensional vector space is that it affords a matrix representation. Since,

$$s_{m+k} = c_0 s_k + c_1 s_{k+1} + \dots + c_{m-1} s_{k+m-1}, k \geq 0. \quad (11)$$

hence a shift register matrix takes the form:

$$T = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & c_0 \\ 1 & 0 & 0 & \dots & 0 & c_1 \\ 0 & 1 & 0 & \dots & 0 & c_2 \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & c_{m-1} \end{bmatrix}$$

and

$$\begin{aligned} (s_{k+1}, s_{k+2}, \dots, s_{k+m}) &= (s_k, s_{k+1}, \dots, s_{k+m-1})T \\ &= (s_{k-1}, s_k, \dots, s_{k-1+m-1})T^2 \\ &= \dots \\ &= (s_0, s_1, \dots, s_{m-1})T^{k+1} \end{aligned}$$

The matrix T is called the state matrix of the LFSR in time domain. Note that $\det(T) = (-1)^m c_0$. Thus T is invertible if and only if $c_0 \neq 0$.

3.2 Frequency Domain Representation of an LFSR Sequence

From (2), DFT of an LFSR sequence a_t defined over a primitive polynomial $f(x)$ produces a Fourier spectrum sequence A_k . For completeness, few important facts are reproduced here from [5] and [21] with some novel observations as well:

1. The zero components in the Fourier spectrum of a sequence over $GF(2^m)$ are related to the roots of a polynomial of that sequence. For example, DFT of an LFSR sequence with feedback polynomial $f(x) = x^3 + x + 1$ initialized with state 001 is $0, 0, 0, \alpha^4, 0, \alpha^2, \alpha$. As roots of $f(x)$ are α along with its conjugates i.e. α^2 and α^4 , so first, second and fourth spectral components are zero.
2. As DFT of a time domain signal comprises of a fundamental frequency and its harmonics, DFT of an LFSR sequence based on a minimal polynomial with no multiple roots also comprises of $\alpha^i \in GF(2^m)$ and its harmonics $\alpha^{ij \bmod r} \in GF(2^m)$ with $0 \leq i \leq r - 1$. This harmonic pattern can be efficiently exploited in cryptanalysis attacks on LFSR based sequence generators.
3. All DFT components of an LFSR sequence $\in GF(2^m)$.
4. indices of non zero DFT points for LFSR with minimum polynomial and no multiple roots also follow a fixed pattern. If k -th component of spectral sequence is non zero then all $(2^j k) \bmod r$ components will be harmonics of the k -th component where $1 \leq j \leq m - 1$.
5. DFT of an LFSR sequence based on a polynomial with multiple roots does not contain harmonic pattern of elements.
6. The Linear Complexity L of an n -periodic sequence is equal to Hamming weight of its frequency domain associate. Three non zero spectral components in example above verifies this fact.

7. **Time Shift Property.** Let two sequences related by a time shift $u_t = s_{t+\tau}$, their DFTs U_k and S_k are related as:

$$U_k = \alpha^{k\tau} S_k, \quad k = 0, 1, \dots, n-1 \quad (12)$$

8. indices of non-zero spectral points of an LFSR sequence does not change with the shift in LFSR sequence. A non-zero k -th component of DFT of an LFSR sequence will always be non-zero. Any shift in LFSR sequence will only change the value at this component by (12). Converse is also true for zero spectral points of an LFSR sequence which will always be zero no matter how much sequence is shifted.
9. **Trace Representation of an LFSR sequence in Frequency Domain.** A binary sequence s_t can be represented in terms of trace function with spectral componenets as follows:-

$$s_t = \sum_{j \in \Gamma(n)} Tr_1^{m_j}(A_j \alpha^{-jt}), \quad t = 0, 1, \dots, n-1 \quad (13)$$

where $Tr_1^{m_j}$ is a trace function from F_{2^m} to F_2 , $A_j \in F_{2^m}$ and $\Gamma(n)$ is a set of cyclotomic coset leaders modulo n .

10. **Matrix Representation in Frequency Domain.** DFT of s_t , being a linear operator with respect to $\alpha \in GF(2^m)$ from equation 2, can be written in matrix form as:

$$(S_0, S_1, S_2, \dots, S_{n-1})^T = D(s_0, s_1, s_2, \dots, s_{n-1})^T \quad (14)$$

where

$$D = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^{2.2} & \dots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \alpha^{n-1} & \alpha^{2(n-1)} & \dots & \alpha^{n-1(n-1)} \end{bmatrix}$$

4 Transform Domain Analysis of a Product Sequence

In this section, transform domain analysis of a product sequence generated through multiplication of two LFSR sequences is presented. This analysis forms the basis for the combiner and filter generators that will appear in subsequent sections. The spectral domain features discussed in Section-3.2 hold true for product sequences as well. In addition, a linear structure existing in the frequency domain representation of the product sequence is presented which renders itself useful for cryptanalysis of LFSR based sequence generators. The initial state of an LFSR has direct relevance to the element $\beta \in GF(2^m)$ in (8) which has been extensively exploited in algebraic and DFT based spectra attacks [19].

Akin to this, there exists another phenomenon which has one to one correspondence with initial states of LFSRs within a linear space $G(f)$ containing cyclic shifts of m sequences. These cyclic shifts in LFSRs sequences and their correspondence to maximum period r possesses certain fixed patterns which exhibit linear behaviour even when employed in nonlinear combiner generators. We have observed that CRT interprets the shifts in LFSRs sequences and is considered as our major contribution to classical theory of LFSRs and sequence generators. As the product of two LFSR sequences is a building block of any non linear boolean function, the idea has been discussed by considering a simple case of two m sequences multiplied together. The process has been generalized through mathematical rationale later in this subsection where CRT based interpretation of shifts in LFSRs sequences has been discussed.

4.1 Time Domain Analysis of a Product Sequence

We build our analysis by starting with a simple case of multiplication of output sequences of two LFSRs and illustrate our novel observations on fixed structures existing in the frequency domain representation of product sequences. The observations of this special case will be generalized to a sequence generator in the next section.

Let s_t be a key stream generated by multiplying the two LFSRs sequences a_t and b_t defined as

$$s_t = f(a_t, b_t) \quad (15)$$

where $f(\cdot)$ is a nonlinear function representing a term wise product. If period of a_t is r_1 and b_t is r_2 , we have

$$s_t = a_i \cdot b_i \quad \text{with } 0 \leq i < r \quad (16)$$

where $r = \text{lcm}(r_1, r_2)$. The linear complexity L of the sequence s_t in this case satisfies

$$L(s_t) \leq L(a_t)L(b_t) \quad (17)$$

where L denotes linear complexity of a sequence and the equality in (17) holds only iff associated polynomials of a_t and b_t are primitive and are greater than 2 [15].

Fixed Patterns in Cyclic Structures of LFSRs It has been observed that there exists a specific relationship between the amount in shifts of product sequence s_t , period of individual LFSRs and shifts from their reference initial states. We generalize the process by giving its mathematical rationale followed by detailed discussion through a small example. CRT allows mathematical representation of relationship observed between the shifts in individual LFSRs, their corresponding periods and overall impact on product sequence s_t . We have a important theorem here.

Theorem 1. *Let $s_t \in GF(2^m)$ be a reference product sequence with period n having two constituent LFSRs defined over primitive polynomials with individual periods n_1 and n_2 . With different shifts k_1 and k_2 in initials states of LFSRs, resulting output sequence u_t is correlated to s_t by (10) where shift τ is determined through CRT as*

$$\begin{aligned}\tau &\equiv k_1 \pmod{n_1} \\ \tau &\equiv k_2 \pmod{n_2}\end{aligned}$$

Proof. Within a cyclic group $GF(2^m)$, associated linear space $G(f)$ of dimension m contains $2^m - 1$ non-zero binary sequences by (9). As s_t and u_t both $\in GF(2^m)$, they are shift equivalents by (10) with unknown shift value of τ .

The product sequence s_t of a_t and b_t can be expressed as

$$s_i = a_j \cdot b_v \tag{18}$$

where $0 \leq i \leq n - 1$, $0 \leq j \leq n_1 - 1$ and $0 \leq v \leq n_2 - 1$.

Remark 1. While contributing towards a product sequence of length n with two LFSRs, stream of LFSR-1 defined over $GF(2^p)$ with primitive polynomial and its maximum period $2^p - 1$ is repeated δ_1 times while LFSR-2 defined over $GF(2^q)$ with primitive polynomial as well and corresponding period $2^q - 1$ is repeated δ_2 where

$$\begin{aligned}\delta_1 &= \frac{\text{lcm}(n_1, n_2)}{n_1}, \text{ and} \\ \delta_2 &= \frac{\text{lcm}(n_1, n_2)}{n_2}\end{aligned}$$

Remark 2. Within a sequence of period n for a product sequence, each value of index j corresponds to all values of index v if and only if $\text{gcd}(n_1, n_2) = 1$.

From Remarks 1 and 2, any shift in LFSRs initial states will produce output corresponding to some fixed indices of j and v which already existed in the reference sequence at some fixed place with initial states of LFSRs without shift.

With known values of j and v i.e. $k_{i's}$, CRT will give us the value of $\tau \bmod n$ as

$$\begin{aligned}\tau &\equiv k_1 \pmod{n_1} \\ \tau &\equiv k_2 \pmod{n_2}\end{aligned}$$

□

Example 1. Let we have a sequence s_t generated from product of two LFSRs having primitive p[olynomials of $g_1(x)x^2 + x + 1$ and $g_2(x)x^3 + x + 1$. The period n_1 of stream a_t corresponding to LFSR-1 is 3 and n_2 of b_t corresponding to LFSR-2 is 7. The period n of s_t is 21.

Table 1 demonstrates product of two m sequences generated from these two LFSRs.

Table 1. Product sequence of 2x LFSRs with $n_1 = 3$ and $n_2 = 7$

Shift Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
a_t	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3
b_t	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7
s_t	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}	s_{21}

We analyze the impact of shift on LFSR sequences and their behaviour in cyclic structures of finite fields involved. We will shift the LFSR sequences one by one and observe the fixed patterns which can be exploited in cryptanalysis of the combiner generators. We can represent shifts in LFSRs sequences with k and l as

$$s_t = a_{i+k} \cdot b_{i+l}, \quad \text{with } 0 \leq i \leq n-1 \quad (19)$$

where $k \in [0, n_1 - 1]$ and $l \in [0, n_2 - 1]$. Table 2 demonstrates the scenerio where a_t is left shifted by one bit while keeping the b_t fixed with initial state of '1'.

Table 2. Product sequence with a_t shifted left

Shift Index	7	8	9	10	11	12	13	14	15	16	17	18	19	20	0	1	2	3	4	5	6
a_t	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1
b_t	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7
u_t	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}	s_{21}	s_1	s_2	s_3	s_4	s_5	s_6	s_7

Comparison of Table 1 with Table 2 reveals that shifting one bit left of a_t and fixing the b_t to reference initial state of '1' shifts s_t by seven units left. Similarly, shifting another bit of a_t to left, brings a_3 corresponding to b_1 which can be located in Table 1 at shift position 14. So two left shifts of a_t shifts s_t by 14 units left with reference to bit positions in Table 1. Now we analyze the impact of left shift of b_t on s_t . Table 3 demonstrates the scenerio where b_t is left shifted by one bit while keeping the a_t fixed with initial state of '1'.

Table 3. Product sequence with b_t shifted left

Shift Index	15	16	17	18	19	20	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
a_t	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3	a_1	a_2	a_3
b_t	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1	b_2	b_3	b_4	b_5	b_6	b_7	b_1
u_t	s_{16}	s_{17}	s_{18}	s_{19}	s_{20}	s_{21}	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{11}	s_{12}	s_{13}	s_{14}	s_{15}

It can be easily seen that one left shift in b_t shifts s_t by 15 units where b_2 is corresponding to a_1 . Similarly, another left shift in b_t shifts s_t by another 15 units bringing the b_3 corresponding to a_1 . Subsequently, three left shifts in b_t with reference to initial state of '1' brings b_4 corresponding to a_1 which is at shift index-3 in Table 1. Similar fixed patterns can be observed for simultaneous shifts of LFSRs and it will be discussed with more detail in following paragraphs.

Let us model this fixed patterns in LFSRs cyclic structures and shifts in initial states of LFSRs through CRT as

$$\begin{aligned} x &\equiv k \pmod{n_1} \\ x &\equiv l \pmod{n_2} \end{aligned}$$

where k and l denote the amount of shifts in initial state of individual LFSRs with reference to initial state of '1'. The solution of CRT i.e. $x \pmod{r}$ gives the amount of shift in s_t with reference to u_t as depicted in (10). Consider a scenerio again where a_t is shifted left by one bit and b_t is fixed with initial state of '1' and can be expressed as

$$\begin{aligned} x &\equiv 1 \pmod{3} \\ x &\equiv 0 \pmod{7} \end{aligned}$$

The CRT gives the solution of $7 \pmod{21}$ which is index position of a_2 corresponding to b_1 in Table 1 shifting the product sequence s_t by seven units left. Consider another scenerio of simultaneous shifts in both LFSRs sequences where a_t is shifted left by one bit and b_t is shifted left by 3 bits with reference to their initial states of '1' and can be expressed as

$$\begin{aligned} x &\equiv 1 \pmod{3} \\ x &\equiv 3 \pmod{7} \end{aligned}$$

The CRT gives value of -11 which is $10 \pmod{21}$, representing the product sequence u_t as 10 units left shifted version of s_t . This value matches to index position of b_4 corresponding to a_2 in Table 1.

Our Observations related to direct correspondence of shift index with initial states of LFSRs and CRT calculations done modulo periods of individual LFSRs are valid for any number of LFSRs in different configurations of nonlinear sequence generators. These observations on classical theory of LFSR cyclic structures with their CRT based interpretation are considered significant for cryptanalysis.

4.2 Frequency Domain Analysis of a Product Sequence

To compute the DFT of the sequence s_t using equation (2), we need to know the minimum polynomial for $s_t \in GF(2^m)$ which can be efficiently determined through Berlekamp-Massey algorithm. It was demonstrated in last subsection that a linear structure exists in the spectral representation of component sequences a_t and b_t which propagates further in the DFT spectra of product sequence s_t . Few interesting results are presented here duly illustrated by an example:

1. Zero and non-zero positions of the DFT Spectra of s_t can be determined even without knowing the minimum polynomial for $s_t \in GF(2^m)$ while working in the lower order associated fields of $a_t \in GF(2^p)$ and $b_t \in GF(2^q)$, where $p \& q < m$. Any k -th component of DFT spectra of s_t is non-zero if and only if A_k and B_k are both non zero, where A_k and B_k represents DFTs of a_t and b_t respectively.
2. With known non-zero DFT points for a_t and b_t , Chinese Remainder Theorem (CRT) can be used to determine non zero points of DFT spectra of s_t directly. With n_1 and n_2 be the individual periods for a_t and b_t respectively, we can apply CRT as:

$$\begin{aligned} x &\equiv k_1 \pmod{n_1} \\ x &\equiv k_2 \pmod{n_2} \end{aligned}$$

where k_1 and k_2 are non zero index positions of A_k and B_k respectively and x is the position of non-zero component of DFT spectra of s_t within its period n .

3. DFT of a product sequence with minimal polynomial having no multiple roots follows a harmonic structure of its elements with $\alpha^i \in GF(2^m)$ and its harmonics $\alpha^{i^j \bmod n} \in GF(2^m)$ with $0 \leq i \leq n-1$, appearing in DFT spectrum.
4. Non-zero indices of DFT sequences also follow a fixed pattern. A k -th non-zero component has its harmonics at all $(2^j k) \bmod n$ points with $1 \leq j \leq n-1$.
5. Shifting of any component sequence a_t or b_t will impact the spectral components of resulting sequence s_t by (12).
6. The zero components in the fourier transform of a product sequence s_t defined over $GF(2^m)$ are related to roots of its minimum polynomial $g(x)$.
7. Consider two LFSR sequences $a_t \in GF(2^p)$ and $b_t \in GF(2^q)$ being components of a product sequence, their spectral components as A_k and B_k respectively with $0 \leq k \leq N-1$. While working in base fields, can we determine their corresponding frequency components of product stream $z_t \in GF(2^m)$? Our results on this problem are being published somewhere else shortly.

Example 2. Consider a product sequence s_t generated from two LFSRs with minimum polynomials $g_1(x) = x^3 + x + 1$ and $g_2(x) = x^2 + x + 1$.

1. In time domain representation, we have following sequences.

Sequence a_t : 011	(of period 3)
Sequence b_t : 0010111	(of period 7)
Sequence s_t : 001011000001010010011101110111011101	(of period 21)
2. From (2), frequency domain representations of these sequences are:
 - (a) $A_k = 0, 1, 1$
 - (b) $B_k = 0, 0, 0, \alpha^4, 0, \alpha^2, \alpha$
 - (c) To compute S_k , associated minimum polynomial is determined through Berlekamp-Massey algorithm which is $g(x) = x^6 + x^4 + x^2 + x + 1$.
 $S_k = 0, 0, 0, 0, 0, \alpha^9, 0, 0, 0, 0, \alpha^{18}, 0, 0, \alpha^{15}, 0, 0, 0, \alpha^{18}, 0, \alpha^9, \alpha^{15}$
3. Following is to be notified here:-
 - (a) Non-zero DFT points in S_k clearly follow a linear behaviour as of time domain representation where any k -th component is non-zero if and only if A_k and B_k are both non-zero. CRT can be directly used to determine these non-zero points.
 - (b) Harmonic pattern of DFT spectra are visible for A_k , B_k and S_k .
 - (c) Non-zero indices of DFT sequences also follow a fixed pattern. In case of S_k , non zero DFT element at index 5 has its harmonics at indices 10, 20, 19 (40 mod 21), 17 (80 mod 21) and at 13 (160 mod 21).
 - (d) If we shift b_t by one bit to left, resulting sequences in frequency domain will hold shift property of equation (12).
 $A_k = 0, 1, 1$
 $B_k = 0, 0, 0, \alpha, 0, \alpha^4, \alpha^2$
 $S_k = 0, 0, 0, 0, 0, \alpha^{18}, 0, 0, 0, 0, \alpha^{15}, 0, 0, \alpha^9, 0, 0, 0, \alpha^{15}, 0, \alpha^{18}, \alpha^9$
 - (e) The zero components in the fourier transform of a product sequence s_t defined over $GF(2^m)$ are related to roots of $g(x) = x^6 + x^4 + x^2 + x + 1$. As roots of $g(x)$ are α alongwith its conjugates i.e. $\alpha^2, \alpha^4, \alpha^8$ and α^{16} so first, second, fourth, eighth and sixteenth spectral components are zero.

The patterns observed in during time and frequency domain analysis of product sequences will be applied to LFSR based sequence generators in the following sections. LFSR based sequence generators can be broadly divided in to three main classes; non-linear filter generators, non-linear combiner generators and clock controlled generators with few variants of shrinking generators [9]. Transform domain anlysis of filter and combiner generators will be presented in following sections.

5 Transformed Domain Analysis of Filter Generators

The nonlinear filter generator consists of a single LFSR which is filtered by a nonlinear boolean function f and is called as filtering function. In a filter generator, the LFSR feedback polynomial, the filtering function and the tapping sequence are usually publicly known. The secret parameter is the initial state of the LFSR which is derived from the secret key of the cipher by a key-loading

algorithm. Therefore, most attacks on filter generators consist of recovering the LFSR initial state from the knowledge of some bits of the sequence produced by the generator (in a known plaintext attack), or of some bits of the ciphertext sequence (in a ciphertext only attack).

5.1 Time Domain Analysis of Filter Generators

Let s_t be an m -sequence with maximum period $2^m - 1$ generated from an LFSR whose length is m , then z_t is the output sequence of a filter generator

$$z_t = f(s_0, s_1, s_2, \dots, s_{t-1}) \quad \forall t \geq 0; \quad (20)$$

where s_0 are inputs of nonlinear function f corresponding to taps of an LFSR as shown in Figure below.

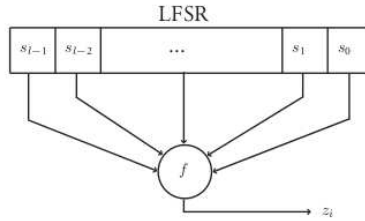


Fig. 1. A Simple Filter Generator

In order to obtain a keystream sequence having good cryptographic properties, the filtering function f should be balanced (i.e., its output should be uniformly distributed), should have large algebraic degree with large correlation and algebraic immunity.

Now we present few important facts related to design parameters of filter generators in time domain [9]:

1. The period of output stream z_t is $2^m - 1$ where m is degree of feedback polynomial of LFSR which is primitive in most cases. The requirement of long period is a direct consequence of shift property of m -sequences as defined in (10) within a linear space $G(f)$.

2. The output sequence of a filter generator z_t is a linear recurring sequence whose linear complexity $L(z)$ is related to length of LFSR and algebraic degree of nonlinear filtering function f . For any LFSR defined over $GF(2)$ with primitive feedback polynomial, an upper bound of linear span is

$$L(z) \leq \sum_{i=1}^d \binom{l}{i} \quad (21)$$

where d is the degree of filtering function f and l is length of LFSR. Thus for a higher L in sequence generator designs, d and l are chosen to be as high as possible.

3. LFSR feedback polynomial should not be sparse.
4. The tapping sequence should be such that the memory size (corresponding to the largest gap between the two taps) is large and preferably close to its maximum possible value. This resists the generalized inversion attack [9] which exploits the memory size.
5. If the LFSR taps are equally spaced, then the lower bounds of linear complexity will be

$$L(z) \geq \binom{l}{d} \quad (22)$$

6. The boolean function used as a filtering function must satisfy following criteria to be called as good cryptographic function. For details, readers may refer to [12].
 - (a) The boolean function should have high algebraic degree. A high algebraic degree increases the linear complexity of the generated sequence.
 - (b) It should have high correlation immunity which is defined as measure of degree to which its output bits are correlated to subset of its input bits. High correlation immunity forces the attacker to consider several input variables jointly and thus decreases the vulnerability of divide-and-conquer attacks.
 - (c) It should have high non-linearity which is related to its minimum distance from all affine functions. A high nonlinearity gives a weaker correlation between the input and output variables. This criteria is in relation to linear cryptanalysis, best affine approximation attacks, and low order approximation attacks. Moreover, high non-linearity resists the correlation and fast correlation attacks [22] by making the involved computations infeasible.
 - (d) The recent developements in cryptanalysis attacks introduced the most significant cryptographic criteria for boolean functions termed as algebraic immunity. There should not be a low degree function g for f which satisfies $f * g = 0$ or $(f + 1) * g = 0$. Algebraic immunity is defined as minimum degree function g for any boolean function f satisfying the said criteria. Attacks exploiting this weakness in boolean functions are called algebraic [11] and fast algebraic attacks [10], [8].
 - (e) A more recent attack on filter generators [25] exploiting the underlying algebraic theory suggests an updated estimate of degree of non linear boolean functions to resist the new kind of attack on filter generators.

5.2 Frequency Domain Analysis of Filter Generators

This section provides DFT based analysis of filter generators which has been known in public domain for few years. Few additional results of our frequency domain analysis has also been included here.

1. Irrespective of time domain requirement of a long period for a filter generator, DFT simplifies the associated high computational problem to any k -th component of Fourier spectra through the relation

$$\alpha^\tau = (Z_k \cdot A_k^{-1})^{k^{-1}} \quad (23)$$

where τ determines the exact amount of shift between a_t and z_t and k is index of any one component of DFT spectra.

2. To obtain specific component of Fourier spectra and limit the computational complexity within affordable bounds, Fast Discrete Fourier spectra attacks [19] propose an idea of selecting a suitable filter polynomial $q(x)$, an LTI system, to pass only specific spectral points and restrict all others which are nulled to zero. To illustrate, we mention an important lemma here without proof.

Lemma 1. *Let $q(x)$ be a polynomial defined over $GF(2)$ with period r as $q(x) = \sum_{i=0}^r c_i x^i$. We apply s_t to LTI system having a function $q(x)$ and z_t is the output sequence. By theory of LTI system response to any arbitrary signal and convolution in time domain, we have*

$$z_t = \sum_{i=0}^r c_i s_{i+t} \quad t = 0, 1, \dots, n-1 \quad (24)$$

Converting the relation into frequency domain, we get

$$Z_k = q(\alpha^k) S_k \quad k = 0, 1, \dots, n-1 \quad (25)$$

where $q(\alpha^k)$ is infact Q_k ; a way of interpreting DFT in terms of polynomials.

3. Selection of polynomial $q(x)$ as a filter function is discussed at length in [17]. To summarize, steps of computations are :-
 - (a) Computing minimum polynomial $g(x)$ of output stream z_t by generating a reference sequence a_t which infact is a shifted version of z_t .
 - (b) Selecting k amongst the coset leaders such that $\gcd(k, n) = 1$ and $g(\alpha^k) = 0$.
 - (c) Computing k -decimated sequence of reference stream a_t as $c_t = a_{kt}$.
 - (d) Using Berlekamp-Massey algorithm, computing minimum polynomial $g_k(x)$.
 - (e) Computing $q(x)$ through a relation

$$q(x) = \frac{g(x)}{g_k(x)} \quad (26)$$

4. Selection of $q(x)$ can be done by direct factorization of minimum polynomial $g(x)$.
5. The DFT over binary fields can be computed for a filter generator without requiring entire sequence. Authors in [19] have provided detailed algorithm for computing DFT for sequences with fewer bits (equal to linear span of that sequence or even lesser) as compared to the total period of the sequence. we also propose an alternate approach to select $q(x)$ in the next section dealing with the combiner generators.
6. The consequence of Fast Discrete Fourier Spectra attacks on sequence generators introduced a new design criterion of spectral immunity. This criteria implies that in order to resist the selective DFT attack, the minimal polynomial of an output sequence of an LFSR based key stream generator should be irreducible.
7. Recent studies have shown that low spectral weight annihilators are essential for Fast Discrete Fourier Spectra attacks [29].

6 Transformed Domain Analysis of Combiner Generators

A combiner generator consists of number of LFSRs which are combined by a nonlinear Boolean function. The Boolean function f is called the combining function and its output is the keystream. The Boolean function f must have high algebraic degree, high nonlinearity and preferably a high order of correlation immunity.

6.1 Time Domain Analysis of Combiner Generators

Consider a combiner generator consisting of l LFSRs as shown in Figure 6.1.

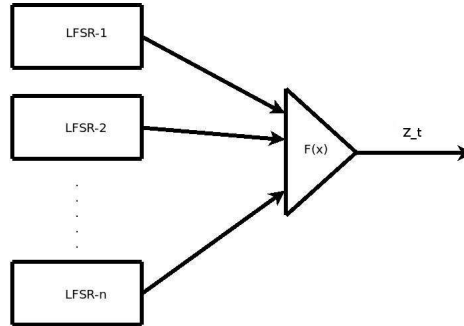


Fig. 2. A Simple Combiner Generator

We denote the output sequence of the i -th LFSR as a^i , its minimal polynomial $g_i(x)$ and its length r_i where $i = 1, 2, \dots, l$. We will assume here that all r_i 's are

mutually coprime which is generally the case for the combiner generators. Let $f(a_t^1, a_t^2, \dots, a_t^l)$ be a non linear function where $f : GF(2^l) \rightarrow GF(2)$ takes input from l LFSRs and produces the key stream z_t as

$$z_t = f(a_t^1, a_t^2, \dots, a_t^l) \quad (27)$$

The criteria for selecting LFSRs and boolean function in a combiner generator are mostly similar to filter generators. However, few additions/differences related to design parameters are as follows:

1. Contrary to filter generators where a single LFSR with larger length is used, combiner generator employs multiple LFSRs with comparatively smaller lengths. The requirement of primitive polynomials as feedback polynomials stays as such for each LFSR.
2. The period of output keystream becomes $lcm(r_1, r_2, \dots, r_l)$.
3. Since the combining function involves both Xoring and multiplication operation, few properties of linear complexities are reproduced here from [7], [26] and [20]. Considering linear complexities of LFSR sequences $a_t^1, a_t^2, \dots, a_t^l$ as L_1, L_2, \dots , and L_l :
 - (a) the linear complexity of the sequence $s_t = (a_t^1 + a_t^2)$ satisfies

$$L(s_t) \leq L(a_t^1) + L(a_t^2), \quad (28)$$

the equality holds if and only if the minimal polynomials of a_t^1 and a_t^2 are relatively prime.

- (b) From (17), the linear complexity of the sequence $s_t = (a_t^1 \cdot a_t^2)$ satisfies

$$L(s_t) \leq L(a_t^1) \cdot L(a_t^2) \quad (29)$$

- (c) The linear complexity of $z_t = f(a_t^1, a_t^2, \dots, a_t^l)$ satisfies

$$L(z_t) = f(L_1, L_2, \dots, L_l) \quad (30)$$

4. The novel observations regarding fixed patterns of LFSRs, cyclic structures existing in finite fields and their interpretation through CRT imply that index of observed keystream bits in any reference stream directly reveals the initial state of all LFSRs. CRT based interpretation of LFSR sequences in relation to their periods thus reiterates the requirement of long period for sequences of combiner generators.

6.2 Frequency Domain Analysis of Combiner Generators

In this section, frequency domain analysis of combiner generators is presented. the application of selective DFT attacks on combiner generators is discussed in detail with some novel observations. Developing on the theory of selective DFT attack, a new efficient methodology is proposed to identify the initial states of all LFSRs of combiner generators.

In section-4.2, the direct mapping of product sequence between time and frequency domains was demonstrated. Likewise, the Xoring operation existing in most of the boolean functions also demonstrates similar mapping. For application of boolean functions in combiner generators, following is important:

1. For the product terms of a non-linear boolean function, a term of DFT spectra of the product stream is nonzero if and only if all the component DFT terms are nonzero. With known non-zero DFT points for $a_t^1, a_t^2, \dots, a_t^l$, CRT can be used to determine non zero points of DFT spectra of s_t directly as:

$$\begin{aligned} x &\equiv k_1 \pmod{r_1} \\ x &\equiv k_2 \pmod{r_2} \\ &\vdots \equiv \dots \dots \dots \\ x &\equiv k_l \pmod{r_l} \end{aligned}$$

where k_1, k_2, \dots, k_l denote non zero index positions of $a_t^1, a_t^2, \dots, a_t^l$ and x is the position of non-zero component of DFT spectra of s_t within its period r .

2. For terms of a non-linear boolean function being Xored, any DFT spectra term will be nonzero for which number of nonzero component DFT terms are odd.

Let us explain these facts through an example.

Example 3. Consider a combiner generator consisting of 3 LFSRs with primitive polynomials as $g_1(x) = x^2 + x + 1$, $g_2(x) = x^3 + x + 1$ and $g_3(x) = x^5 + x^2 + 1$. The outputs of LFSRs in this case are m-sequences, denoted as a_t^1, a_t^2 and a_t^3 respectively. Output stream of the generator is denoted as z_t and a nonlinear function $f(x)$ is

$$f(x) = a_t^1.a_t^2 + a_t^2.a_t^3 + a_t^1.a_t^3$$

where period of z_t in this case becomes 651 as $\text{lcm}(3, 7, 31) = 651$.

Mapping of operations is demonstrated in spectral domain using 2 as:

1. DFT of individual LFSRs:
 - (a) DFT of a_t^1 : 0, 1, 1
 - (b) DFT of a_t^2 : 0, 0, 0, α^4 , 0, α^2 , α
 - (c) DFT of a_t^3 : only five non-zero DFT terms are at indices 15, 23, 27, 29 and 30 with values $\alpha^{29}, \alpha^{30}, \alpha^{15}, \alpha^{23}$ and α^{27} respectively.
2. DFT of product streams:
 - (a) DFT of $a_t^1 a_t^2$: Corresponding to minimum polynomial of $x^6 + x^4 + x^2 + x + 1$ and period = 21, six non-zero components are $\alpha^9, \alpha^{18}, \alpha^{15}, \alpha^{18}, \alpha^9$ and α^{15} at indices 5, 10, 13, 17, 19 and 20. These indices can be easily determined while working in component fields of $GF(2^2)$ and $GF(2^3)$ by using CRT calculations as discussed in Section 4.2.

- (b) DFT of $a_t^2 a_t^3$: With minimum polynomial $x^{10} + x^5 + x^4 + x^2 + 1$ and period = 93, ten non-zero components are at indices 23, 29, 46, 58, 61, 77, 85, 89, 91 and 92.
- (c) DFT of $a_t^3 a_t^1$: Similarly with minimum polynomial $x^{15} + x^{12} + x^{10} + x^7 + x^6 + x^2 + 1$ and period = 217, fifteen non-zero components are at indices 27, 54, 61, 89, 108, 122, 139, 153, 178, 185, 201, 209, 213, 215 and 216.
- (d) To verify the established facts, DFT of product of all three streams has also been analyzed. For $a_t^1 a_t^2 a_t^3$ having a minimum polynomial of $x^{30} + x^{25} + x^{24} + x^{20} + x^{19} + x^{17} + x^{16} + x^{13} + x^{10} + x^9 + x^8 + x^7 + x^4 + x^2 + 1$ with period = 651, thirty non-zero DFT components are at indices 61, 89, 122, 139, 178, 185, 209, 215, 244, 271, 278, 325, 356, 370, 395, 418, 430, 433, 461, 488, 523, 542, 556, 587, 619, 635, 643, 647, 649 and 650. All these indices can be determined directly from knowing the individual DFTs of three LFSRs separately. For instance,

$$\begin{aligned} x &\equiv 1 \pmod{3} \\ x &\equiv 3 \pmod{7} \\ x &\equiv 15 \pmod{31} \end{aligned}$$

gives result of 325 which exists amongst thirty non-zero DFT computations as well.

3. To see impact of Xor operation in frequency domain, DFT of z_t with minimum polynomial $x^{31} + x^{29} + x^{28} + x^{27} + x^{24} + x^{23} + x^{22} + x^{20} + x^{18} + x^{17} + x^{16} + x^{15} + x^{13} + x^{11} + x^{10} + x^9 + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$ is computed. Results reveal that all indices where number of non-zero DFT terms for three product streams i.e. $a_t^1 a_t^2$, $a_t^2 a_t^3$ and $a_t^1 a_t^3$ are odd, resulting DFT term is non-zero. For instance indices at 5, 10, 13, 17, 19 and 20 where only DFT of $a_t^1 a_t^2$ term is non-zero, resulting DFT for z_t is also non-zero. Similar is the case for other indices.

Selective DFT Attacks on Combiner Generators. In this subsection, possibility of extending selective DFT attack on combiner generators has been discussed. The attack algorithm on non-linear filter generator has been explained in [19] and [17]. However, direct application of selective DFT attack on combiner generators has few limitations with regard to underlying design of these type of sequence generators. For simplicity, case of $m = L(a_t)$ has been considered here where m is the number of known bits of key stream and a_t is the coordinated scaled sequence of key stream.

1. As combiner generators entail multiple LFSRs, determination of element $\beta \in GF(2^m)$ through coordinated scaled sequence doesnot lead to initial states of all LFSRs directly.
2. In precomputation stage of the algorithm as discussed at length in [17], k -decimation sequence of LFSR output sequence is computed followed by applying Berlekamp-Massey algorithm on it to determine its associated minium

polynomial $g_k(x)$. With the help of this sequence c_t , $m \times m$ circulant matrix is obtained as follows

$$M = \begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_{m-2} & c_{m-1} \\ c_1 & c_2 & c_3 & \dots & c_{m-1} & c_m \\ c_2 & c_3 & c_4 & \dots & c_m & c_{m+1} \\ \cdot & \cdot & \cdot & \dots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot & \cdot \\ c_{m-1} & c_m & c_{m+1} & \dots & c_{2m-3} & c_{2m-2} \end{bmatrix}$$

3. In [19], this matrix is termed as coefficient matrix H and $g_k(x)$ is minimum polynomial for α^k where a_t is determined from (8) or through frequency component as:

$$a_t = \sum_k Tr_1^{n_k}(A_k \alpha^{tk}) \quad t = 0 \dots n-1 \quad (31)$$

4. We propose another approach to compute $g_k(x)$ through factoring $g(x)$. In this case, step involving Berlekamp-Massey algorithm on k -decimated sequence will no longer be required. Setting up matrix M or H in this case is by direct initializing of corresponding LFSR from any random state.
5. The output of selective DFT algorithm produces $\beta = \alpha^\tau$. Using this value of τ , left shift value for each LFSR sequence is determined by applying modular computations of CRT with respect to individual periods r_i 's of LFSRs as:

$$\begin{aligned} \tau_1 &\equiv \tau \pmod{r_1} \\ \tau_2 &\equiv \tau \pmod{r_2} \\ &\dots \dots \dots \\ \tau_l &\equiv \tau \pmod{r_l} \end{aligned}$$

6. Determine the initial state of each LFSR by applying individual shift values τ_i 's to LFSRs sequences by using (8) within each field $GF(2^m)$ where

$$\begin{aligned} \beta_i &= \alpha^{\tau_i} \\ b_t^i &= Tr_1^n(\beta_i \alpha_i^t) \end{aligned}$$

7. If τ is directly applied to each LFSR, number of computations involved in shifting LFSR sequence is of the order to $\mathcal{O}(\tau)$. CRT based interpretation of LFSR shifts in initial states with respect to their periods save the last step computations of selective DFT attacks.

Let us demonstrate these observations through an example.

Example 4. Consider the same combiner generator as in Example 3. Suppose we have only 31 bits of keystream $z_t = [1011110001111010111001011010111]$. With a known structure of the generator, our attack will determine the initial state of three LFSRs as follows:

1. Initially, possibility of success of selective DFT attack on given combiner generator will be determined. Through Berlekamp-Massey algorithm, minimum polynomial $g(x)$ of keystream z_t will be computed. Applying factorization algorithm on $g(x)$ gives its three factors as $g_1(x) = x^6 + x^4 + x^2 + x + 1$, $g_2(x) = x^{10} + x^5 + x^4 + x^2 + 1$ and $g_3(x) = x^{15} + x^{12} + x^{10} + x^7 + x^6 + x^2 + 1$. So the selective DFT attack on this combiner generator is possible.
2. Generating a reference sequence a_t and decimating it with $k = 58$ with $\gcd(58, 651) = 1$ produces a coefficient sequence $c_t = 011010101110$.
3. Applying Berlekamp-Massey algorithm on c_t gives its associated minimum polynomial of $g_k(x) = x^6 + x^4 + x^2 + x + 1$.
4. Circulant matrix will thus be of dimension 6×6 as:

$$M = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}$$

5. From (26), filter polynomial $q(x) = x^{25} + x^{22} + x^{19} + x^{17} + x^{10} + x^9 + x^8 + x^5 + 1$.
6. The same results could be achieved from our observations as follows:
 - (a) From DFTs of a_t^1 , a_t^2 and a_t^3 while working in finite fields of $GF(2^3)$, $GF(2^3)$ and $GF(2^5)$, $k = 58$ can be directly calculated to be non-zero index for DFT of z_t .
 - (b) Factorization done initially to check applicability of our selective DFT attacks already showed $g_1(x) = g_k(x)$ as one of the factor.
 - (c) We thus obtain the same $q(x)$ as in step-5 above.
7. Computations of selective DFT algorithm give the result of $\tau = 19$.
8. Finally, we can determine the left shift value in sequences for each LFSR by applying modular computations of CRT with respect to individual periods of LFSRs r_i 's as

$$\tau_1 \equiv 19 \pmod{3}$$

$$\tau_2 \equiv 19 \pmod{7}$$

$$\tau_l \equiv 19 \pmod{31}$$

9. Having determined the exact shift value for each LFSR, their initial states will be computed by using (8) within each field $GF(2^m)$ where

$$\begin{aligned} \beta_i &= \alpha^{\tau_i} \\ b_t^i &= Tr_1^n(\beta_i \alpha_i^t) \end{aligned}$$

10. The initial fills of LFSRs with 1 left shifts in a_t^1 , 5 left shifts in a_t^2 and 19 left shifts in a_t^3 gives:

Table 4. Initial States of 3 LFSRs

	Initial State
LFSR-1	10
LFSR-2	101
LFSR-3	01111

We are thus able to recover exactly the initials states of all the LFSRs through our novel approach of interpreting fixed patterns in LFSRs sequences through CRT. Although the proposed approach was demonstrated on an illustrative example, but it holds true for any configuration of combiner generators. The same was tested with different non-linear combiner functions as well as with different number of LFSRs.

6.3 Complexity Comparisons

Let us see the advantage of frequency domain analysis of LFSR based combiner generator over its time domain analysis. We will discuss computational complexity in relation to most common attacks also. For a combiner sequence having period N with l number of constituent LFSRs, complexity of Exhaustive search is $2^{m_1+m_2+\dots+m_l-1}$ where as for correlation attack complexity reduces to $2^{m_1-1} + 2^{m_2-1} + \dots + 2^{m_l-1}$. To compute the complexity in case of selective DFT attacks on combiner generators, calculations for preprocessing and actual attack stage are [19]:

1. Preprocessing Stage. The computations during this stage are sum of following:
 - (a) The complexity of computing minimum polynomial $g(x)$ by applying Berlekamp-Massey algorithm on s_t is $\mathcal{O}(L \log_2(L))$ Xor operations where L is linear complexity of s_t . Incase of using a relation $g(x) = \prod_{k \in I} g_k(x)$, complexity is $|I|[\log_2(L)]^3$.
 - (b) The complexity of computing $g_k(x)$ is $m \log_2(m)^2$ for each k . For $k \in I$ with I representing set of coset leaders, complexity will be $|I|[m (\log_2 m)^2]$.
 - (c) The complexity of computing $g(\alpha^k)$ is $\mathcal{O}(N \eta(m) + m \log_2(N))$ GF(2) operations where $\eta(m) = (m \log m \log \log m)$ and N is the degree of polynomial $g(x)$.
2. Attack Stage. The complexity of this stage is sum of following:
 - (a) Passing our sequence s_t from LTI filter $q(x)$ is actually time convolution of $q(x)$ and s_t which costs $\mathcal{O}(L)$ GF(2) operations.
 - (b) Last step of DFT spectra attack giving the output β is solving the system of linear equations over GF(2) in d unknowns which has the complexity of utmost $\mathcal{O}(d^w)$, where w is Strassen's reduction exponent $w = \log_2(7) \approx 2.807$.
 - (c) Determining left shift values for each LFSR through $\tau_i \equiv \tau \bmod r_i$ and computing initial states of LFSRs are of order $\mathcal{O}(l)$ each and are negligible, where l is number of LFSRs.

Having established the comparisons of complexities between exhaustive search, correlation attack and selective DFT attack, let us map these to our example-3 above. Exhaustive search costs utmost $2^{2+3+5-1} = 2^9 \approx \mathcal{O}(512)$ $GF(2)$ computations. With Probabilities of $P(z = a) = 6/8$, $P(z = b) = 6/8$ and $P(z = c) = 6/8$, correlation attack largely reduces the complexity to $2^{2-1} + 2^{3-1} + 2^{5-1} \approx \mathcal{O}(21)$ $GF(2)$ operations. Incase of selective DFT attacks, complexity of preprocessing stage is $\mathcal{O}(279)$ and of attack stage is $\mathcal{O}(150)$ making total of $\mathcal{O}(429)$ $GF(2)$ operations. Thus it can be clearly stated that after one time preprocessing computations of selective DFT attacks, complexity of actual attack stage with $\mathcal{O}(150)$ is promisingly less as compared to exhaustive search attack. However, correlation attacks and their faster variants are more efficient than DFT attacks in special scenerios where underlying combining function is not correlation immune. Incase of correlation immune combining functions, selective DFT attacks still provide propitious results and are advantageous over the correlation attacks.

7 Applicability of Fast Discrete Fourier Spectra Attacks on A5/1 Algorithm

In this section applicability of fast discrete fourier spectra attacks on A5/1 algorithm is discussed. For clarity of context description of algorithm structure is given first followed by discussion on possibility of selective DFT attacks on it.

7.1 Description of A5/1

A5/1 is a stream cipher built on a clock controlled combiner generator. Being a Global System for Mobile Communications (GSM) encryption algorithm, it has been intensively analyzed and is considered weak because of number of successful attacks now. Details can be found in [4], [13], and [14]. The keystream generator consists of three LFSRs, $R1$, $R2$ and $R3$ of lengths 19, 22 and 23 respectively as shown in figure-3.

The taps of three LFSRs correspond to primitive polynomials $x^{19} + x^5 + x^2 + x + 1$, $x^{22} + x^1 + 1$ and $x^{23} + x^{15} + x^2 + x + 1$ and therefore, LFSRs produce maximum periods. The registers are clocked irregularly based on decision of a majority function having input of clocking bits 8, 10 and 10 of registers $R1$, $R2$ and $R3$ respectively. It is a type of stop and go clocking where those LFSRs are clocked whose most significant bit (msb) matches to output bit of the majority function. At each clock cycle either two or three registers are clocked, and that each register is moved with probability 3/4 and stops with probability 1/4. The running key of A5/1 is obtained by XORing of the output of the three LFSRs. The process of keystream generation is as follows:-

1. Starting with key initialization phase, all LFSRs are initialized to zero first. 64-bit session key $K = (k_0, \dots, k_{63})$ and publically known 22-bit frame number serve as initialization vector. In this phase all three registers are clocked

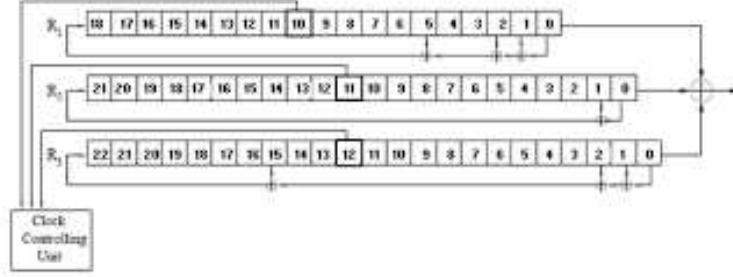


Fig. 3. Design of A5/1 Algorithm

regularly for 86 cycles during which the key bits followed by frame bits are xored with the least significant bits of all three registers consecutively.

2. During second phase, the three registers are clocked for 100 additional cycles with the irregular clocking, but the output is discarded.
3. Finally, the generator is clocked for 228 clock cycles with the irregular clocking producing the 228 bits that form the keystream. 114 of them are used to encrypt uplink traffic from A to B , while the remaining 114 bits are used to decrypt downlink traffic from B to A . A GSM conversation is sent as a sequence of frames, where one frame is sent every 4.6 ms and contains 114 bits. Each frame conversation is encrypted by a new session key K .

7.2 DFT Attacks on A5/1

Applicability of selective DFT attacks is preconditioned with following two separate cases:-

1. Case 1. If the minimal polynomial of output keystream z_t is reducible, algorithm 1 described in [19] is directly applicable.
2. Case 2. If another sequence $y_t \in GF(2^n)$ is determined such that $v_t = z_t * y_t$, where $*$ is a term wise product, and $L(v_t) + L(y_t) \leq L(z_t)$ and $L(v_t + y_t) + L(y_t) \leq L(z_t)$, algorithm 2 described in [19] is applicable on z_t .

Our analysis reveals that selective DFT attacks on A5/1 algorithm are not applicable. Detailed results are being published at other forum shortly

7.3 DFT Attacks on E0 Cipher

Selective DFT attacks on E0 cipher are possible with modifications in equations derived in [3]. Our results on E0 cipher are being published at some other forums shortly.

8 Conclusion

In this report, we presented a transform domain analysis of LFSR based sequence generators. The inherent peculiarities of the LFSR product sequences were evoked through novel patterns identified with the help of a CRT based approach. These findings were then extended to the filter generators and more particularly to the combiner generators. An effort was made to establish the mapping of different operations from time domain to frequency domain. Novel results on fixed shift patterns of LFSRs, their relationship to cyclic structures in finite fields and CRT based interpretation of these patterns have been exploited to reduce the computations required in the last stage of DFT spectral attacks on combiner generators. Subsequent to the transform domain analysis of basic components of stream ciphers and discussion on applicability of fast discrete fourier attacks on A5/1 algorithm, DFT based analysis of combiners resistant to correlation attacks are considered as interesting cases for their analysis in frequency domain and some initial results have shown good promise in this regard.

References

1. <http://www.ecrypt.eu.org/stream/>, April 2014.
2. <http://www.etsi.org/services/security-algorithms/cellular-algorithms>, May 2014.
3. Frederik Armknecht. A linearization attack on the bluetooth key stream generator. 2002.
4. Elad Barkan and Eli Biham. Conditional estimators: An effective attack on a5/1. In *Selected Areas in Cryptography*, pages 1–19. Springer, 2006.
5. RE Blahut. *Theory and practice of error control codes*. Addison-Wesley Publishing Company, USA, 1983.
6. CIG Bluetooth. Specification of the bluetooth system, version 1.1 (february 22, 2001).
7. Lennart Brynielsson. On the linear complexity of combined shift register sequences. In *Advances in CryptologyEUROCRYPT85*, pages 156–160. Springer, 1986.
8. Anne Canteaut. Open problems related to algebraic attacks on stream ciphers. In *Coding and cryptography*, pages 120–134. Springer, 2006.
9. Anne Canteaut. Stream cipher. *Encyclopedia of Cryptography and Security*, pages 1263–1265, 2011.
10. Nicolas T Courtois. Fast algebraic attacks on stream ciphers with linear feedback. In *Advances in Cryptology-CRYPTO 2003*, pages 176–194. Springer, 2003.
11. Nicolas T Courtois and Willi Meier. Algebraic attacks on stream ciphers with linear feedback. In *Advances in CryptologyEUROCRYPT 2003*, pages 345–359. Springer, 2003.
12. Thomas W Cusick and Pantelimon Stanica. *Cryptographic Boolean functions and applications*. Academic Press, 2009.
13. Patrik Ekdahl and Thomas Johansson. Another attack on a5/1. *Information Theory, IEEE Transactions on*, 49(1):284–289, 2003.
14. Timo Gendrullis, Martin Novotný, and Andy Rupp. A real-world attack breaking a5/1 within hours. In *Cryptographic Hardware and Embedded Systems-CHES 2008*, pages 266–282. Springer, 2008.

15. Solomon W Golomb and Guang Gong. *Signal design for good correlation: for wireless communication, cryptography, and radar*. Cambridge University Press, New York, USA, 2005.
16. Solomon Wolf Golomb, Lloyd R Welch, Richard M Goldstein, and Alfred W Hales. *Shift register sequences*, volume 78. Aegean Park Press Laguna Hills, CA, 1982.
17. Guang Gong. A closer look at selective dft attacks.
18. Guang Gong and Solomon W Golomb. Transform domain analysis of des. *Information Theory, IEEE Transactions on*, 45(6):2065–2073, 1999.
19. Guang Gong, Sondre Rønjom, Tor Helleseth, and Honggang Hu. Fast discrete fourier spectra attacks on stream ciphers. *Information Theory, IEEE Transactions on*, 57(8):5555–5565, 2011.
20. Tore Herlestam. On functions of linear shift register sequences. In *Advances in CryptologyEUROCRYPT85*, pages 119–129. Springer, 1986.
21. James L Massey and Shirlei Serconek. A fourier transform approach to the linear complexity of nonlinearly filtered sequences. In *Advances in Cryptology-CRYPTO94*, pages 332–340. Springer, 1994.
22. Willi Meier and Othmar Staffelbach. Fast correlation attacks on certain stream ciphers. *Journal of Cryptology*, 1(3):159–176, 1989.
23. John M Pollard. The fast fourier transform in a finite field. *Mathematics of computation*, 25(114):365–374, 1971.
24. Sondre Rønjom, Guang Gong, and Tor Helleseth. On attacks on filtering generators using linear subspace structures. In *Sequences, Subsequences, and Consequences*, pages 204–217. Springer, 2007.
25. Sondre Rønjom and Tor Helleseth. A new attack on the filter generator. *Information Theory, IEEE Transactions on*, 53(5):1752–1758, 2007.
26. R Rueppel and Othmar Staffelbach. Products of linear recurring sequences with maximum complexity. *Information Theory, IEEE Transactions on*, 33(1):124–131, 1987.
27. Rainer A Rueppel. *Analysis and design of stream ciphers*. Springer-Verlag New York, Inc., 1986.
28. Peter Sweeney. *Error Control Coding: from theory to practice*. John Wiley and Sons, Ltd., West Sussex PO19 1UD, England, 2002.
29. Jingjing Wang, Kefei Chen, and Shixiong Zhu. Annihilators of fast discrete fourier spectra attacks. In *Advances in Information and Computer Security*, pages 182–196. Springer, 2012.